Katholieke Universiteit Leuven

Departement Elektrotechniek ESAT-SISTA/TR 1996-52I

Closed-Loop Subspace System Identification [1]

Peter Van Overschee and Bart De Moor[2]

September 1996 - Revision 1 June 199
Submitted to IFAC Automatica

[1] This report is available by anonymous ftp from *ftp.esat.kuleuven.ac.be* in the directory */pub/SISTA/vanoverschee/reports/clos.ps.Z.* This work is supported by: Concerted Research Action GOA-MIPS (Model-based Information Processing Systems), the FWO project G.0256.97: Numerical Algorithms for Subspace System Identification, extension to special cases, the FWO Research Communities: ICCoS (Identification and Control of Complex Systems) and Advanced Numerical Methods for Mathematical Modeling, the Interuniversity Poles of Attraction Programme (IUAP P4-02 (1997-2001): Modeling, Identification, Simulation and Control of Complex Systems; and IUAP P4-24 (1997-2001): Intelligent Mechatronic Systems (IMechS)), the Human Capital and Mobility Network SIMONET (System Identification and Modeling Network), SCIENCE-ERNSI (European Research Network for System Identification): SC1-CT92-0779. The scientific responsibility is assumed by its authors.

[2] ESAT(SISTA) - Katholieke Universiteit Leuven, Kardinaal Mercierlaan 94, 3001 Leuven (Heverlee), Belgium, Tel 32/16/32 10 95, Fax 32/16/32 19 70, WWW: *http://www.esat.kuleuven.ac.be/sista.* E-mail: *peter.vanoverschee@esat.kuleuven.ac.be.* and *bart.demoor@esat.kuleuven.ac.be.* Peter Van Overschee is a senior research assistant and Bart De Moor a senior research associate of the Belgian National Fund for Scientific Research.

Abstract

In this paper we present a general framework for closed-loop subspace system identification. This framework consists of two new projection theorems which allow the extraction of non-steady state Kalman filter states and of system related matrices directly from closed-loop input output data. Three algorithms for the identification of the state space matrices are then presented. The similarities between the theorems and algorithms, and the corresponding open-loop theorems and algorithms in the literature are emphasized. The closed-loop theory is illustrated with a simulation example.

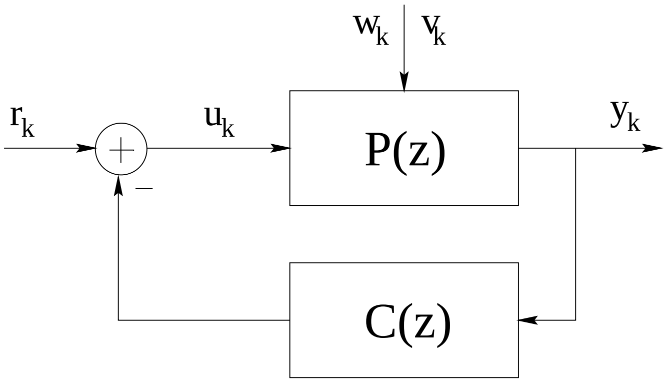

Figure 1: Standard feedback setup. u_k is the input signal, y_k the output signal and r_k the reference signal. v_k (measurement noise) and w_k (process noise) are the disturbances acting on the linear plant $P(z)$. The linear controller is represented by $C(z)$. In this paper we assume that u_k and y_k are measured and that a limited number of impulse response samples (Markov parameters) of the controller is known. The controller $C(z)$ does not have to be explicitly given.

1 Introduction

Subspace identification methods have been proven to be a valuable alternative for classical prediction error or instrumental variables methods [10, 13]. Different breads of subspace identification methods have been published [8, 15, 20] of which a good overview is presented in [4, 18, 21]. All of these subspace methods have a number of nice properties in common: unlike classical methods, they do not suffer from the problems caused by a-priori parametrizations, initial estimates and non-linear optimizations. They also identify multivariable systems in a very simple and elegant way. The trade-off is that the obtained models are sub-optimal, in the sense that (up to now) it is not known whether they are the optimal solution to an error-minimization criterion.

In practice it is often necessary to perform identification experiments on systems operating in closed-loop. This is especially true when open-loop experiments are not allowed due to safety (unstable processes) or production (undesirable open-loop behavior) reasons. System identification from closed-loop data is thus a relevant topic.

All of the published subspace identification methods however have one major drawback, they do not work when the data is gathered under closed-loop experimental conditions as in Figure 1. Indeed, when there is feedback, the results of any of the algorithms [8, 15, 20] are asymptotically biased [11]. This is also true for the Canonical Variate Analysis, despite the claim in the introduction of [9]. In this paper we show how this problem can be solved, i.e. how subspace algorithms have to be modified in order to compute asymptotically unbiased results.

For an inventory of possible solutions to the closed-loop identification problem, we refer to [5]. In this introduction, we only summarize the contributions that have been published relating to closed-loop *subspace* identification.

In [11] it is analyzed why subspace methods do not work in closed-loop. An interpretation of subspace methods in the framework of k-step ahead predictors is given. This interpretation leads to a possible solution to overcome the closed-loop subspace identification problems.

However, the problem is the estimation of the k-step ahead predicted outputs, for which a high order ARX model is used. This method leads to biased results and deviates quite significantly from published subspace algorithms. Moreover for this method to be asymptotically unbiased, the ARX models from which the predictors are constructed, have to be infinitely dimensional. In [19] a subspace method is used to jointly identify the transfer from reference input r_k to process input u_k and output y_k (see Figure 1). From this joint model, a model of the plant and of the controller is then distilled. This method is very general and any identification method could be used for the identification of the joint model. The advantage of using a subspace identification method is the inherent MIMO nature of the problem (since u_k and y_k are jointly identified as outputs, there is always more than one output) and the fact that the separation of the plant and controller is simplified when working with state space systems since a part of the model reduction can be done analytically.

Even though this method is general applicable and leads to asymptotically unbiased results, there are a couple of drawbacks. First of all, there is the problem of identifying the joint input/output process. Indeed, the jointly identified model has an order equal to the sum of the plant and controller order. If the order of the controller is high, the identification problem at hand becomes unnecessary complicated. A second drawback is the extra model reduction step that has to be applied in the second step of the algorithm. Finally there is the inherent problem of unstable pole-zero cancelations, which restricts the controller and/or the plant to be minimum phase and/or stable [19].

In this paper, we assume that, just as in [11], the signals u_k, y_k are measured (compare to [19] where u_k, y_k and r_k have to be measured). Furthermore, we assume that a limited number (see further) of impulse response samples (Markov parameters) of the controller are given (no controller information is needed in [11] nor in [19] where the controller is identified). So knowledge of the controller itself is not needed, and it does not need to be identified. This is a clear advantage when the controller itself is fairly complicated and/or of high order. The only knowledge required is the limited number of impulse response samples, which can be easily obtained from a simple measurement on the digital controller. If this impulse response measurement is not feasible, the needed impulse response samples can be derived (identified) from the given signals, but then the reference signal r_k needs to be given.

Given the above, the contributions of this paper are the following:

- We derive a general framework for the identification of systems operating in closed-loop. This framework consists of two theorems, which are very similar to the theorems derived for the open-loop case. Moreover, when the controller goes to zero, these theorems reduce to the open-loop theorems of [15, 18]. This is a desirable feature, since it implies that the open-loop algorithms of [15, 18] are just special cases of this more general closed-loop theory.

  So, instead of reducing the closed-loop problem to an open-loop one as in [19], we incorporate from the first step of the algorithm the knowledge we have about the controller into the algorithm, which leads to a general theory.

- We present an interpretation of the state-sequence for closed loop measurements. Indeed, the theorems in this paper show how the non-steady state Kalman filter sequence of the open-loop system ($P(z)$ in Figure 1) can be derived in closed loop from an oblique projection of the data matrices. This result is again very similar to the results of [15, 18] where it was shown how an oblique projection determines the non-steady state Kalman filter states for open-loop systems.

- Using the derived theorems, we present three algorithms, which are, once again, very similar to the open-loop algorithms presented in the literature. Indeed, with the controller set to zero, the algorithms in this paper correspond to algorithms in the literature. Much attention is also paid to the asymptotic properties of the algorithms.

  The algorithms handle stable as well as unstable plants and controllers, and there is no restriction on the zeros of the controller nor the plant.

- With a simple simulation example we illustrate the asymptotic properties of the new algorithms and compare them with open and closed-loop algorithms of the literature.

This paper is organized as follows. In Section 2 we present the problem description and the notation in more detail. Section 3 contains the general closed-loop framework with the two main theorems. The algorithms are described in Section 4. All concepts are illustrated with an example in Section 5. The conclusions and further results are stated in Section 6.

2 Problem Description and Notation

In this section we introduce the closed-loop identification problem, the notation used for block Hankel matrices, for projections and for some matrix operations.

The identification setup considered in this paper is the following: $u_k \in \mathbb{R}^m, y_k \in \mathbb{R}^l$ and $r_k \in \mathbb{R}^m$ are the input, output and reference[1] signal generated by the closed-loop configuration of Figure 1. The signals are connected through the following state space systems.

- The plant equations:

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + w_k \ , & (1)\\ y_k &= Cx_k + Du_k + v_k \ , & (2) \end{aligned}$$

$$\mathbf{E}[\begin{pmatrix} w_k \\ v_k \end{pmatrix} \begin{pmatrix} w_l^t & v_l^t \end{pmatrix}] = \begin{pmatrix} Q & S \\ S^t & R \end{pmatrix} \delta_{kl} \quad \geq 0^2 \ , \qquad (3)$$

  and $A, Q \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times m}, C \in \mathbb{R}^{l\times n}, D \in \mathbb{R}^{l\times m}, S \in \mathbb{R}^{n\times l}$ and $R \in \mathbb{R}^{l\times l}$. $v_k \in \mathbb{R}^l$ and $w_k \in \mathbb{R}^n$ are unobserved, zero mean, white noise vector sequences. $\{A, C\}$ is assumed to be observable, while $\{A, [\ B \ \ Q^{1/2}\ ]\}$ is assumed to be controllable. Note that A should not be stable, which it indeed often is not in a closed-loop measurement situation. The plant transfer matrix $P(z)$ is given by:

$$P(z) = D + C(zI_n - A)^{-1}B \quad \in \quad \mathbb{C}^{l\times m} \ , \qquad (4)$$

  with I_n the n dimensional identity matrix. In exactly the same way as described in [15, 18] the combined system (1)-(2) is split up in a deterministic and a stochastic

[1] Note that r_k is an m dimensional vector while y_k is l dimensional. r_k should thus be regarded as the reference signal filtered through a feedforward block with l inputs and m outputs. This is the two degrees of freedom controller. See also Section 4.5 and Figure 4 for a one degree of freedom configuration.

[2] $\mathbf{E}$ denotes the expected value operator and δ_{kl} the Kronecker delta.

subsystem. y_k^d, x_k^d and y_k^s, x_k^s denote the deterministic respective stochastic output and state[3]. For the stochastic subsystem, we have the following definitions (see also [14, 18]):

$$\begin{aligned}
\Sigma_{ss} &\stackrel{\text{def}}{=} \mathbf{E}[x_k^s.(x_k^s)^T] \ , \\
G &\stackrel{\text{def}}{=} \mathbf{E}[x_{k+1}^s.(y_k^s)^T] \ , \\
\Lambda_0 &\stackrel{\text{def}}{=} \mathbf{E}[y_k^s.(y_k^s)^T] \ .
\end{aligned}$$

- The controller equations:

$$x_{k+1}^c = A_c x_k^c + B_c y_k \ , \tag{5}$$

$$u_k = r_k - C_c x_k^c - D_c y_k \ , \tag{6}$$

  with $A_c \in \mathbb{R}^{n_c \times n_c}, B_c \in \mathbb{R}^{n_c \times l}, C_c \in \mathbb{R}^{m \times n_c}, D_c \in \mathbb{R}^{m \times l}$. The controller transfer matrix $C(z)$ is given by:

$$C(z) = D_c + C_c(zI_{n_c} - A_c)^{-1}B_c \quad \in \quad \mathbb{C}^{m \times l} \ . \tag{7}$$

  Note that the controller should not be stable. We denote the Markov parameters of the controller by $\mathcal{C}_i = C_c A_c^{i-1} B_c, i \neq 0$ and $\mathcal{C}_0 = D_c$.

We assume that the closed-loop identification problem is well-posed, in a sense that the output y_k is uniquely determined by the states of the plant and the controller and by the disturbances and reference input. This generic condition is satisfied when $I_l + DD_c$ is non-singular.

The problem treated in this paper can now be stated as:

Given:

- Input and output measurements of a well-posed problem: $u_k, y_k,\ k = 0, \ldots, j + 2i - 2$ where $j \to \infty$
- The first i Markov parameters of the controller $C(z)$: $\mathcal{C}_0, \ldots \mathcal{C}_{i-1}$

Find:

- The system order n
- The system matrices A, B, C, D up to within a similarity transformation
- The matrices Q, S, R up to within second order statistics.

Before presenting the solution to this problem, we need to define some system related matrices. The extended observability matrices are defined as:

$$\Gamma_i \stackrel{\text{def}}{=} \begin{pmatrix} C \\ CA \\ \ldots \\ CA^{i-1} \end{pmatrix} \in \mathbb{R}^{li \times n} \quad , \quad \Gamma_i^c \stackrel{\text{def}}{=} \begin{pmatrix} C_c \\ C_c A_c \\ \ldots \\ C_c A_c^{i-1} \end{pmatrix} \in \mathbb{R}^{mi \times n_c} \ . \tag{8}$$

[3] Throughout the paper the sub or superscript “s” stands for “stochastic, “d” for “deterministic” and “c” for “controller”.

The reversed extended controllability matrices are defined as:

$$\Delta_i^d \overset{\text{def}}{=} \begin{pmatrix} A^{i-1}B & \dots & AB & B \end{pmatrix} \in \mathbb{R}^{n\times mi} \ , \tag{9}$$

$$\Delta_i^c \overset{\text{def}}{=} \begin{pmatrix} A_c^{i-1}B_c & \dots & A_cB_c & B_c \end{pmatrix} \in \mathbb{R}^{n_c\times li} \ , \tag{10}$$

$$\Delta_i^G \overset{\text{def}}{=} \begin{pmatrix} A^{i-1}G & \dots & AG & G \end{pmatrix} \in \mathbb{R}^{n\times li} \ . \tag{11}$$

We define the lower triangular block-Toeplitz matrices H_i^d and H_i^c as:

$$H_i^d \overset{\text{def}}{=} \begin{pmatrix} D & 0 & \dots & 0 \\ CB & D & \dots & 0 \\ \dots & \dots & \dots & \dots \\ CA^{i-2}B & CA^{i-3}B & \dots & D \end{pmatrix} \in \mathbb{R}^{li\times mi} \ , \tag{12}$$

$$H_i^c \overset{\text{def}}{=} \begin{pmatrix} D_c & 0 & \dots & 0 \\ C_cB_c & D_c & \dots & 0 \\ \dots & \dots & \dots & \dots \\ C_cA_c^{i-2}B_c & C_cA_c^{i-3}B_c & \dots & D_c \end{pmatrix} \in \mathbb{R}^{mi\times li} \ . \tag{13}$$

The lower triangular block-Toeplitz matrix T_i is defined as:

$$T_i \overset{\text{def}}{=} (I_{li} + H_i^d H_i^c)^{-1} \in \mathbb{R}^{li\times li} \ , \tag{14}$$

where we assumed that the matrix in (14) is invertible, which is the case when the closed-loop identification problem is well posed, i.e. when $I_l + DD_c$ is invertible.

Input block Hankel matrices are defined as:

$$U_{0|i-1} \overset{\text{def}}{=} \begin{pmatrix} u_0 & u_1 & u_2 & \dots & u_{j-1} \\ u_1 & u_2 & u_3 & \dots & u_j \\ \dots & \dots & \dots & \dots & \dots \\ u_{i-1} & u_i & u_{i+1} & \dots & u_{i+j-2} \end{pmatrix} \in \mathbb{R}^{mi\times j} \ ,$$

where we assume that $j \to \infty$ throughout the paper. The index i is user-defined and it should be "large enough", i.e. be larger than the order of the system to be identified. The subscripts of U denote the subscript of the first and last element of the first column. Similar definitions hold for the output block Hankel matrices $Y_{0|i-1}$, the stochastic output block Hankel matrices $Y^s_{0|i-1}$ and the reference block Hankel matrices $R_{0|i-1}$. We will make extensive use of the following data matrices, constructed from the input and output block Hankel matrices[4]:

$$N_{p|q} \overset{\text{def}}{=} \left[U_{0|2i-1} + H_{2i}^c Y_{0|2i-1}\right]_{\text{block rows p through q}} \ , \tag{15}$$

$$M_{p|q} \overset{\text{def}}{=} U_{p|q} + H_{q-p+1}^c Y_{p|q} \ , \tag{16}$$

where $0 \leq p \leq q \leq 2i-1$. Note that:

- None of the matrices $N_{p|q}$ or $M_{p|q}$ are block Hankel matrices.

[4] Note that we only need the input and output signals and the first Markov parameters of the controller to generate these matrices.

- The following relation between $N_{p|q}$ and $M_{p|q}$ is easy to establish:

$$\begin{aligned} N_{p|q} &= U_{p|q} + H^c_{q-p+1} Y_{p|q} + \Gamma^c_{q-p+1} \Delta^c_p Y_{0|p-1} \\ &= M_{p|q} + \Gamma^c_{q-p+1} \Delta^c_p Y_{0|p-1} \ . \end{aligned} \tag{17}$$

- $M_{i+1|2i-1}$ is not equal to $M_{i|2i-1}$ without the first block column (denoted with $\overline{M}_{i|2i-1}$). Indeed, it is easy to show that:

$$M_{i+1|2i-1} = \overline{M}_{i|2i-1} - \Gamma^c_{i-1} B_c Y_{i|i} \ . \tag{18}$$

The state sequences are defined as:

$$X_i = \begin{pmatrix} x_i & x_{i+1} & \dots & x_{i+j-1} \end{pmatrix} \ , \tag{19}$$

$$X^c_i = \begin{pmatrix} x^c_i & x^c_{i+1} & \dots & x^c_{i+j-1} \end{pmatrix} \ , \tag{20}$$

$$X^d_i = \begin{pmatrix} x^d_i & x^d_{i+1} & \dots & x^d_{i+j-1} \end{pmatrix} \ , \tag{21}$$

$$X^s_i = \begin{pmatrix} x^s_i & x^s_{i+1} & \dots & x^s_{i+j-1} \end{pmatrix} \ . \tag{22}$$

In subspace identification algorithms, we typically assume for theoretical analysis that there are long time series available ($j \to \infty$). Due to the assumed ergodicity, we can replace the expected value operator $\mathbf{E}$ (average over different experiments) with the operator $\mathbf{E_j}$ (average over one, infinitely long experiment) [18]:

$$\mathbf{E_j}[\bullet] \overset{\text{def}}{=} \lim_{j \to \infty} \frac{1}{j} [\bullet]$$

With this definition of $\mathbf{E_j}$, the correlation matrices are then defined as:

$$\mathbf{E_j} \left[\begin{pmatrix} N_{0|2i-1} \\ X_0 \end{pmatrix} \begin{pmatrix} N^T_{0|2i-1} & X^T_0 \end{pmatrix} \right] \overset{\text{def}}{=} \begin{pmatrix} R_{nn} & S^T_{xn} \\ S_{xn} & \Sigma_{xx} \end{pmatrix} \ , \tag{23}$$

$$\mathbf{E_j} \left[\begin{pmatrix} Y^s_{0|i-1} \\ Y^s_{i|2i-1} \end{pmatrix} \begin{pmatrix} (Y^s_{0|i-1})^T & (Y^s_{i|2i-1})^T \end{pmatrix} \right] \overset{\text{def}}{=} \begin{pmatrix} L_i & (\Delta^G_i)^T \Gamma^T_i \\ \Gamma_i \Delta^G_i & L_i \end{pmatrix} \ , \tag{24}$$

$$\mathbf{E_j} \left[\begin{pmatrix} X^d_0 \\ X^s_0 \end{pmatrix} \begin{pmatrix} (X^d_0)^T & (X^s_0)^T \end{pmatrix} \right] \overset{\text{def}}{=} \begin{pmatrix} \Sigma_{dd} & \Sigma^T_{sd} \\ \Sigma_{sd} & \Sigma_{ss} \end{pmatrix} \ . \tag{25}$$

Apart from the state sequences in (19)-(22) we also introduce the concept of the Kalman filter state sequences. For an elaborate discussion, we refer the reader to [15, 18]. Here, we will only introduce the definition:

Definition 1 Kalman filter states
For a given symmetric $P_0 \in \mathbb{R}^{n \times n}$ and initial state sequence matrix $X^k_0 \in \mathbb{R}^{n \times j}$, the Kalman filter state sequence X^k_i is defined as:

$$\begin{aligned} X^k_{i\,[X^k_0, P_0]} &= \begin{pmatrix} \widehat{x}_i & \widehat{x}_{i+1} & \dots & \widehat{x}_{i+j-1} \end{pmatrix} \\ &\overset{def}{=} (A - \Omega_i \Gamma_i) X^k_0 + (\Delta^d_i - \Omega_i H^d_i) U_{0|i-1} + \Omega_i Y_{0|i-1} \ , \end{aligned} \tag{26}$$

$$\Omega_i \overset{def}{=} (\Delta^G_i - A^i P_0 \Gamma^T_i)(L_i - \Gamma_i P_0 \Gamma^T_i)^{-1} \tag{27}$$

Each state $\widehat{x}_p$ can be interpreted as the result of a non-steady state Kalman filter running on the data in the corresponding column of $U_{0|i-1}$ and $Y_{0|i-1}$. The initial state estimate of this bank of Kalman filters is contained in X_0^k while the error covariance matrix on these state initial estimates is equal to:

$$\widetilde{P}_0 = \Sigma_{ss} - P_0 \ . \tag{28}$$

We refer the reader to [15, 18] for more background information.

The following geometric operations are defined as in [18]:

- The orthogonal projection $\mathcal{A}/\boldsymbol{\mathcal{B}}$ of the row space of the matrix $\mathcal{A} \in \mathbb{R}^{p \times j}$ on the row space of the matrix $\mathcal{B} \in \mathbb{R}^{q \times j}$ is defined as the row space of the matrix:

$$\mathcal{A}/\boldsymbol{\mathcal{B}} \overset{\text{def}}{=} \mathbf{E_j}[\mathcal{A}\mathcal{B}^T] \left[\mathbf{E_j}[\mathcal{B}\mathcal{B}^T]\right]^{\dagger} \mathcal{B} \ , \tag{29}$$

  where $\bullet^{\dagger}$ denotes the Moore-Penrose pseudo-inverse of the matrix $\bullet$.

- The oblique projection $\mathcal{A}/_{\mathcal{B}}\boldsymbol{\mathcal{C}}$ of the row space of $\mathcal{A} \in \mathbb{R}^{p \times j}$ along the row space of $\mathcal{B} \in \mathbb{R}^{q \times j}$ on the row space of $\mathcal{C} \in \mathbb{R}^{r \times j}$ is defined as the row space of the matrix:

$$\mathcal{A}/_{\mathcal{B}}\boldsymbol{\mathcal{C}} \overset{\text{def}}{=} \mathbf{E_j}[\mathcal{A}\begin{pmatrix} \mathcal{C}^T & \mathcal{B}^T \end{pmatrix}] \left[\mathbf{E_j}\begin{pmatrix} \mathcal{C}\mathcal{C}^T & \mathcal{C}\mathcal{B}^T \\ \mathcal{B}\mathcal{C}^T & \mathcal{B}\mathcal{B}^T \end{pmatrix}\right]^{\dagger} \begin{pmatrix} I_r \\ 0 \end{pmatrix} \mathcal{C} \ . \tag{30}$$

  Important properties of the oblique projection operator are:

$$\mathcal{B}/_{\mathcal{B}}\boldsymbol{\mathcal{C}} = 0 \qquad , \qquad \mathcal{C}/_{\mathcal{B}}\boldsymbol{\mathcal{C}} = \mathcal{C} \ . \tag{31}$$

For a more detailed description and a geometric interpretation, we refer to [18].

3 A Closed-Loop Framework

In this section we derive a framework for closed-loop subspace system identification. We first briefly indicate what the typical problem is when using a classical subspace identification algorithm in closed-loop. Next we discuss two new closed-loop projection theorems that show how the non-steady state Kalman filter state sequences and system related matrices can be retrieved from the data matrices. In a next section, these insights will then lead to general subspace algorithms which are valid for open as well as closed-loop system identification.

Subspace algorithms [8, 18, 20] are typically derived for open-loop situations. In contrast with the conjectures of Larimore [9] who claims that his Canonical Variate Algorithm “works” in closed-loop, open-loop subspace algorithms applied to closed-loop situations lead to biased results. The main reason that open-loop subspace algorithms do not work for signals measured in closed-loop is that the disturbances v_k, w_k and the input u_k are correlated. Indeed, each of the algorithms in [8, 18, 20] critically relies on the assumption that the disturbances and the inputs are orthogonal to each other. These facts were already pointed out and proven in [11] so we will not prove them here again. The simulation example in Section 5 will clearly illustrate that open-loop subspace algorithms indeed compute biased results in closed-loop.

3.1 closed-loop projection Theorem

We can get around the problem of the computation of biased results through the introduction the new data matrices N and M as in (15)-(16). Indeed, unlike the input data matrices, which are correlated with the disturbances v_k and w_k, these new data matrices are uncorrelated with these disturbances. This fact facilitates the derivation of the two new theorems in the section. The first theorem relates the non-steady state Kalman filter states to an orthogonal projection of data matrices.

Theorem 1 Closed-Loop projection Theorem
Under the assumption that:

1. *The reference signal r_k is uncorrelated with the process noise w_k and measurement noise v_k.*
2. *The matrix $N_{0|2i-1}$ has full row rank $2mi$.*
3. *There is an infinite amount of measurements available : $j \to \infty$.*
4. *The closed-loop problem is well posed, i.e. $I_l + DD_c$ is invertible.*

Then:

$$\begin{aligned} \mathcal{Z}_i &\overset{def}{=} Y_{i|2i-1} / \begin{pmatrix} \boldsymbol{U}_{0|i-1} \\ \boldsymbol{Y}_{0|i-1} \\ \boldsymbol{M}_{i|2i-1} \end{pmatrix} \\ &= Y_{i|2i-1} / \begin{pmatrix} \boldsymbol{N}_{0|2i-1} \\ \boldsymbol{Y}_{0|i-1} \end{pmatrix} && (32) \\ &= T_i \left[\Gamma_i \widehat{X}_i + H_i^d M_{i|2i-1} \right] , && (33) \end{aligned}$$

with:

$$\begin{aligned} \widehat{X}_i &= X_i^k{}_{[\widehat{X}_0, P_0]} , && (34) \\ \widehat{X}_0 &= X_0 / \boldsymbol{N}_{0|2i-1} , && (35) \\ P_0 &= -\left[\Sigma_{xx} - \Sigma_{ss} - S_{xn} R_{nn}^{-1} S_{xn}^T \right] . && (36) \end{aligned}$$

The proof of the closed-loop Theorem can be found in Appendix A.

Remarks and discussion:

- The second assumption in the Theorem corresponds to the regular persistency of excitation assumption used in subspace identification. However, instead of using the input data matrix $U_{0|2i-1}$ we use the transformed data matrix $N_{0|2i-1}$.
- The matrix $\widehat{X}_i$ can, just as in the open-loop case, be interpreted as the state sequence estimated by a bank of non-steady state Kalman filters [15, 18]. This bank of Kalman

filters has the following initial state sequence $\widehat{X}_0$ and initial error covariance matrix $\tilde{P}_0$ (see (28)):

$$\begin{aligned} \widehat{X}_0 &= X_0/\boldsymbol{N}_{0|2i-1} \ , \\ \widetilde{P}_0 &= \Sigma_{ss} - P_0 \\ &= \Sigma_{xx} - S_{xn}R_{nn}^{-1}S_{xn}^T \\ &= \mathbf{E_j}\left[(X_0 - \widehat{X}_0)(X_0 - \widehat{X}_0)^T\right] \ , \end{aligned}$$

which indicates that the error on the initial state $\widetilde{P}_0$ is given by the variance of the part of the "real" state X_0 which is not contained in the row space of $N_{0|2i-1}$.

- This Theorem resembles very much the typical projection Theorems in for instance [15, 18, 20], where it is stated that, for open-loop measurements, the following holds:

  $$Y_{i|2i-1}/\begin{pmatrix} \boldsymbol{U}_{0|i-1} \\ \mathbf{Y}_{0|i-1} \\ \boldsymbol{U}_{i|2i-1} \end{pmatrix} = \Gamma_i\widehat{X}_i + H_i^dU_{i|2i-1} \ .$$

  Comparing this to (33) we see that there are however three differences between this Theorem and the classical open-loop input-output equations:

  - The matrix T_i in (33)
  - The inputs $U_{i|2i-1}$ are replaced by the matrix $M_{i|2i-1}$ in (33)
  - The initial state $\widehat{X}_0 = X_0/\boldsymbol{U}_{0|2i-1}$ in the open-loop case is replaced by the initial state $\widehat{X}_0 = X_0/\boldsymbol{N}_{0|2i-1}$ in (35).

3.2 closed-loop main Theorem

Just as for the open-loop identification problems, we present a main subspace identification theorem. This theorem allows for the computation of the row space of a Kalman filter state sequence and of the column space of the product of T_i and the extended observability matrix $(T_i\Gamma_i)$, directly from the input-output data, without any knowledge of the system matrices.

Theorem 2 Closed-Loop main Theorem
Under the same assumptions as in Theorem 1 and with $\mathcal{O}_i$ defined as the oblique projection:

$$\mathcal{O}_i \overset{def}{=} Y_{i|2i-1}/_{M_{i|2i-1}}\begin{pmatrix} \boldsymbol{U}_{0|i-1} \\ \mathbf{Y}_{0|i-1} \end{pmatrix} \ , \tag{37}$$

and the singular value decomposition:

$$\mathcal{O}_i = \begin{pmatrix} U_1 & U_2 \end{pmatrix}\begin{pmatrix} S_1 & 0 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix} = U_1S_1V_1^T \ , \tag{38}$$

we have:

1. *The oblique projection $\mathcal{O}_i$ equals T_i times the extended observability matrix times a Kalman filter state sequence:*
$$\mathcal{O}_i = T_i \Gamma_i . \widetilde{X}_i \ , \tag{39}$$
with:
$$\begin{aligned} \widetilde{X}_i &\overset{def}{=} X^k_{i\,[\widetilde{X}_0, P_0]} \ , \\ \widetilde{X}_0 &= \widehat{X}_0 /_{M_{i|2i-1}} \begin{pmatrix} \boldsymbol{U}_{0|i-1} \\ \boldsymbol{Y}_{0|i-1} \end{pmatrix} \ , \\ P_0 &= -\left[\Sigma_{xx} - \Sigma_{ss} - S_{xn} R_{nn}^{-1} S_{xn}^T \right] \ . \end{aligned} \tag{40}$$

2. *The order of the system (1)-(2) is equal to the number of singular values in equation (38) different from zero.*

3. *The product of T_i and the extended observability matrix can be taken equal to:*
$$T_i \Gamma_i = U_1 S_1^{1/2} \ . \tag{41}$$

4. *The state sequence $\widetilde{X}_i$ can be taken equal to:*
$$\begin{aligned} \widetilde{X}_i &= S_1^{1/2} V_1^T \\ &= (T_i \Gamma_i)^\dagger \mathcal{O}_i \ . \end{aligned} \tag{42}$$

The proof of this Theorem follows directly from Theorem 1 and from the properties of the oblique projection (31).

Remarks:

- Theorem 2 can algebraically be summarized as:

$$\boxed{\begin{array}{rcl} \text{rank} \ (Y_{i|2i-1} /_{M_{i|2i-1}} \begin{pmatrix} \boldsymbol{U}_{0|i-1} \\ \boldsymbol{Y}_{0|i-1} \end{pmatrix}) & = & n \\ \text{row space} \ (Y_{i|2i-1} /_{M_{i|2i-1}} \begin{pmatrix} \boldsymbol{U}_{0|i-1} \\ \boldsymbol{Y}_{0|i-1} \end{pmatrix}) & = & \text{row space} \ (\widetilde{X}_i \,) \\ \text{column space} \ (Y_{i|2i-1} /_{M_{i|2i-1}} \begin{pmatrix} \boldsymbol{U}_{0|i-1} \\ \boldsymbol{Y}_{0|i-1} \end{pmatrix}) & = & \text{column space} \ (T_i \Gamma_i \,) \end{array}}$$

  This summary is the essence to label this algorithm as a *subspace* algorithm: they retrieve system related quantities via subspaces of projected data matrices.

- It should be noted that the state sequence $\widetilde{X}_i$ recovered through this theorem differs from the state sequence $\widehat{X}_i$ introduced in Theorem 1. The two sequences are indeed different due to their different initial state (35) and (40).

- Once again, this Theorem is very similar to the oblique projection theorems of [15, 18]. Indeed, these theorems state that, for open-loop measurements, the oblique projection:

  $$Y_{i|2i-1}/_{U_{i|2i-1}} \begin{pmatrix} \boldsymbol{U}_{0|i-1} \\ \boldsymbol{Y}_{0|i-1} \end{pmatrix} = \Gamma_i \widetilde{X}_i \; .$$

  Once again, the difference with (37) consist of the matrix T_i, the use of $M_{i|2i-1}$ instead of $U_{i|2i-1}$ and the different expression for $\widetilde{X}_0$.

4 Closed-Loop Subspace Identification Algorithms

In this section we explain how the different system matrices A, B, C, D and Q, R, S can be estimated from Theorem 1 and/or 2. It is worth noting that the first part of the algorithms is equivalent for all of the algorithms and coincides with the first steps described in Theorem 2 (the oblique projection and Singular Value Decomposition).

4.1 Algorithm 1: unbiased, using the states

From Theorem 2 we find the order n of the system (step 2) and the product $T_i\Gamma_i$ (step 3). The following side result from Theorem 1 can easily be proven.

$$\begin{aligned}
\mathcal{Z}_{i+1} &= Y_{i+1|2i-1}/\begin{pmatrix} \boldsymbol{U}_{0|i} \\ \boldsymbol{Y}_{0|i} \\ \boldsymbol{M}_{i+1|2i-1} \end{pmatrix} \\
&= Y_{i+1|2i-1}/\begin{pmatrix} \boldsymbol{N}_{0|2i-1} \\ \boldsymbol{Y}_{0|i} \end{pmatrix} && (43) \\
&= T_{i-1}\left[\Gamma_{i-1}\widehat{X}_{i+1} + H^d_{i-1}M_{i+1|2i-1}\right] \; , && (44) \\
\widehat{X}_{i+1} &\overset{\text{def}}{=} X^k_{i+1[\widehat{X}_0, P_0]} \; , && (45)
\end{aligned}$$

where $\widehat{X}_0$ and P_0 are given by (35) and (36).

For now, imagine that $\widehat{X}_i$ and $\widehat{X}_{i+1}$ are given. Since they are both Kalman filter state sequences of the same non-steady state Kalman filter bank[5], one can prove that:

$$\begin{aligned}
\widehat{X}_{i+1} &= A\widehat{X}_i + BU_{i|i} + K_iE_{i|i} \; , && (46) \\
Y_{i|i} &= C\widehat{X}_i + DU_{i|i} + E_{i|i} \; , && (47) \\
E_{i|i} &= Y_{i|i} - C\widehat{X}_i - DU_{i|i} \; . && (48)
\end{aligned}$$

where K_i is the non-steady state Kalman gain after i iterations and $E_{i|i}$ contains the corresponding innovations [15, 18]. These equations are very similar to the equations obtained for the open-loop situation. There is however one subtle difference between (46)-(48) and the open-loop Kalman filter equations: the innovation $E_{i|i}$ are *not* orthogonal to the inputs $U_{i|i}$.

[5] The Kalman filters are the same in the sense that they have the same initial estimate $\widehat{X}_0$ and the same initial error covariance $\widetilde{P}_0$.

So, given the state $\widehat{X}_i$ and $\widehat{X}_{i+1}$, it is not possible to determine the system matrices from equations (46)-(47) with a simple least squares algorithm.
Through the use of the results in Theorem 1 it can however be proven that the innovations $E_{i|i}$ in (46)-(48) are perpendicular to the row space of $\widehat{X}_i$ and $M_{i|i}$. For a proof we refer the reader to Appendix B. In this Appendix it is also shown that (46)-(47) can be rewritten as:

$$\left(\frac{\widehat{X}_{i+1}}{Y_{i|i}}\right) = \left(\begin{array}{c|c} A - BD_cT_1C & B(I_m - D_cT_1D) \\ \hline T_1C & T_1D \end{array}\right)\left(\frac{\widehat{X}_i}{M_{i|i}}\right) + \left(\frac{K_i - BD_cT_1}{T_1}\right) E_{i|i} \ , \quad (49)$$

with (from 14) $T_1 = (I_l + DD_c)^{-1}$ non-singular since the problem is well posed. Given $\widehat{X}_i$ and $\widehat{X}_{i+1}$, equation (49) can now be solved with a simple least squares algorithm, since the residuals $E_{i|i}$ are perpendicular to the regressors $\widehat{X}_i$ and $M_{i|i}$. Given $D_c(= \mathcal{C}_0)$, the system matrices A, B, C, D and Q, R, S can then be easily unraveled from the least squares solution and residuals (see Figure 2).

The only remaining question to be solved is how to determine the Kalman filter state sequences $\widehat{X}_i$ and $\widehat{X}_{i+1}$ from $\mathcal{Z}_i$ and $\mathcal{Z}_{i+1}$. In order to achieve this, we first have to determine the matrix $T_iH_i^d$ in (33). With U_2 from the singular value decomposition (38), we find from (33):

$$U_2^T \mathcal{Z}_i = U_2^T.(T_iH_i^d).M_{i|2i-1} \ . \quad (50)$$

We now consider the elements in the lower triangular block Toeplitz matrix $T_iH_i^d$ as unknowns:

$$T_iH_i^d \overset{\text{def}}{=} \mathcal{K} \overset{\text{def}}{=} \begin{pmatrix} \mathcal{K}_0 & 0 & \ldots & 0 \\ \mathcal{K}_1 & \mathcal{K}_0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ \mathcal{K}_{i-1} & \mathcal{K}_{i-2} & \ldots & \mathcal{K}_0 \end{pmatrix} \ , \quad (51)$$

with $\mathcal{K}_k \in \mathbb{R}^{l \times m}$. To solve $\mathcal{K}_k$ from this, one can either multiply (50) from the right with $M_{i|2i-1}^\dagger$ and then solve for the unknowns by solving a simple set of linear equations. Our experience however is that, when $M_{i|2i-1}$ is not so well conditioned, it is better to solve (50) directly for the unknowns without multiplication with $M_{i|2i-1}^\dagger$. We describe in Appendix C how to do this. Once $\mathcal{K}$ is known, it is easy to prove that (from (33) and (44) and with $\mathcal{G} = T_i\Gamma_i = U_1S_1^{1/2}$):

$$\begin{aligned} \widehat{X}_i &= \mathcal{G}^\dagger \left[\mathcal{Z}_i - \mathcal{K}.M_{i|2i-1}\right] \ , & (52) \\ \widehat{X}_{i+1} &= \underline{\mathcal{G}}^\dagger \left[\mathcal{Z}_{i+1} - \underline{\mathcal{K}|}.M_{i+1|2i-1}\right] \ , & (53) \end{aligned}$$

where $\underline{\mathcal{G}}$ indicates $\mathcal{G}$ without the last block row and $\underline{\mathcal{K}|}$ denotes $\mathcal{K}$ without the last block row and column.
The final algorithm described in this section is summarized in Figure 2. It computes asymptotically unbiased estimates of the system matrices A, B, C, D and biased estimates of Q, R, S, the bias on which is due to the fact that the Kalman filters are non-steady state, i.e. in equation (49) we have a non-steady state Kalman gain K_i. This bias disappears when $i \to \infty$.

This algorithm is very similar to the unbiased algorithm in [15, 18]. The major difference with the algorithm in this paper is that here the matrix $\mathcal{K}$ is explicitly determined. In [15, 18] this

closed-loop algorithm 1:

1. Construct the matrices $M_{i|2i-1} = U_{i|2i-1} + H_i^c Y_{i|2i-1}$ and $M_{i+1|2i-1} = U_{i+1|2i-1} + H_{i-1}^c Y_{i+1|2i-1}$ where H_i^c and H_{i-1}^c contain the given Markov parameters of the controller.

2. Calculate the oblique and orthogonal projections :

$$\mathcal{O}_i = Y_{i|2i-1} /_{M_{i|2i-1}} \begin{pmatrix} \boldsymbol{U}_{0|i-1} \\ \mathbf{Y}_{0|i-1} \end{pmatrix} ,$$

$$\mathcal{Z}_i = Y_{i|2i-1} / \begin{pmatrix} \boldsymbol{U}_{0|i-1} \\ \mathbf{Y}_{0|i-1} \\ \boldsymbol{M}_{i|2i-1} \end{pmatrix} \quad , \quad \mathcal{Z}_{i+1} = Y_{i+1|2i-1} / \begin{pmatrix} \boldsymbol{U}_{0|i} \\ \mathbf{Y}_{0|i} \\ \boldsymbol{M}_{i+1|2i-1} \end{pmatrix} .$$

3. Calculate the **SVD** of the oblique projection $\mathcal{O}_i = USV^T$.

4. Determine the order by inspecting the singular values in S and partition the **SVD** accordingly to obtain U_1, U_2 and S_1.

5. Determine $\mathcal{G} = U_1 S_1^{1/2}$ and $\mathcal{K}$ as in Appendix C.

6. Determine the states:

$$\begin{aligned} \widehat{X}_i &= \mathcal{G}^\dagger \left[\mathcal{Z}_i - \mathcal{K} M_{i|2i-1} \right] , \\ \widehat{X}_{i+1} &= \underline{\mathcal{G}}^\dagger \left[\mathcal{Z}_{i+1} - \underline{\mathcal{K}|} M_{i+1|2i-1} \right] . \end{aligned}$$

7. Solve the set of linear equations for the solution $\mathcal{S}$ and the residuals $\mathcal{T}$:

$$\begin{pmatrix} \widehat{X}_{i+1} \\ Y_{i|i} \end{pmatrix} = \begin{pmatrix} \mathcal{S}_{11} & \mathcal{S}_{12} \\ \mathcal{S}_{21} & \mathcal{S}_{22} \end{pmatrix} \begin{pmatrix} \widehat{X}_i \\ M_{i|i} \end{pmatrix} + \mathcal{T} .$$

   with $M_{i|i} = U_{i|i} + D_c Y_{i|i}$.

8. Determine $A, B, C, D, (T_1)$ as ($\mathcal{C}_0$ is given): $B = \mathcal{S}_{12}(I_m - \mathcal{C}_0 \mathcal{S}_{22})^{-1}$, $D = \mathcal{S}_{22}(I_m - \mathcal{C}_0 \mathcal{S}_{22})^{-1}$, $A = \mathcal{S}_{11} + B\mathcal{C}_0 \mathcal{S}_{21}$, $T_1 = (I_l + D\mathcal{C}_0)^{-1}$, $C = T_1^{-1} \mathcal{S}_{21}$.

9. Determine Q, S and R from the residuals as:

$$\begin{pmatrix} Q & S \\ S^T & R \end{pmatrix} = \begin{pmatrix} I_n & B\mathcal{C}_0 \\ 0 & T_1^{-1} \end{pmatrix} \mathbf{E_j}[\mathcal{T}\mathcal{T}^T] \begin{pmatrix} I_n & B\mathcal{C}_0 \\ 0 & T_1^{-1} \end{pmatrix}^T$$

Figure 2: A schematic overview of the first closed-loop subspace identification algorithm.

is not needed since the $\widehat{X}_i$ and $\widehat{X}_{i+1}$ are implicitly determined from the orthogonal projection theorem (Theorem 1). One could try something similar in the closed-loop case, i.e. one could, without the explicit determination of $\mathcal{K}$, write the states $\widehat{X}_i$ and $\widehat{X}_{i+1}$ as (52) and (53) and substitute these equations in (49). This leads to:

$$\begin{pmatrix} \underline{\mathcal{G}}^\dagger \mathcal{Z}_{i+1} \\ Y_{i|i} \end{pmatrix} = \begin{pmatrix} \mathcal{L}_{11} \\ \mathcal{L}_{21} \end{pmatrix} \mathcal{G}^\dagger \mathcal{Z}_i + \begin{pmatrix} \mathcal{L}_{12} \\ \mathcal{L}_{22} \end{pmatrix} M_{i|2i-1} + \text{residuals} \ . \tag{54}$$

With $U_{i|2i-1}$ replaced by $M_{i|2i-1}$, this equation corresponds exactly to the open-loop equations found in [15, 18]. However, in the open-loop case, if we solve (54) in a least squares sense, we find that $\mathcal{L}_{11} = A$ and $\mathcal{L}_{21} = C$. As is shown in Appendix D, for the closed-loop case, we get as a solution from (D):

$$\begin{aligned} \mathcal{L}_{11} &= A - BD_cT_1C - \underline{\mathcal{G}}^\dagger T_{i-1}H^d_{i-1}\Gamma^c_{i-1}B_cT_1C \ , & (55) \\ \mathcal{L}_{21} &= T_1C \ . & (56) \end{aligned}$$

The third term in (55) indicates that this is not a good way to proceed. So, even if the analogy with open-loop identification is appealing, it does not go through completely.
Note however that the algorithm described in Figure 2 works perfectly well for open-loop data. Indeed, in the open-loop case, $H^c_i = 0$ and $\mathcal{C}_0 = 0$. The matrices T_i, T_{i-1} and T_1 thus become identity matrices and the matrix $\mathcal{K}$ becomes equal to H^d_i. Finally, we find that, for the open-loop case, $\mathcal{S}_{11} = A, \mathcal{S}_{12} = B, \mathcal{S}_{21} = C, \mathcal{S}_{22} = D$.

4.2 Algorithm 2: biased, using the states

As stated before, the sequence $\widehat{X}_i$ can not be directly computed from the projections, and we need to compute the matrix $\mathcal{K}$ to find these state sequences (52)-(53). However, from Theorem 2 we find that the state sequence $\widetilde{X}_i$ *can* be determined directly from the oblique projection $\mathcal{O}_i$. Similarly, the state sequence $\widetilde{X}_{i+1}$ can be determined from the oblique projection:

$$\begin{aligned} \mathcal{O}_{i+1} &= Y_{i+1|2i-1} /_{M_{i+1|2i-1}} \begin{pmatrix} \boldsymbol{U}_{0|i} \\ \boldsymbol{Y}_{0|i} \end{pmatrix} \\ &= T_{i-1}\Gamma_{i-1}\widetilde{X}_{i+1} \ . & (57) \end{aligned}$$

And we thus find $\widetilde{X}_i$ and $\widetilde{X}_{i+1}$. The problem however is that both Kalman filter sequences have a different initial state (40):

$$\begin{aligned} \text{Initial state for } \widetilde{X}_i &= \widehat{X}_0 /_{M_{i|2i-1}} \begin{pmatrix} \boldsymbol{U}_{0|i-1} \\ \boldsymbol{Y}_{0|i-1} \end{pmatrix} \\ \text{Initial state for } \widetilde{X}_{i+1} &= \widehat{X}_0 /_{M_{i+1|2i-1}} \begin{pmatrix} \boldsymbol{U}_{0|i} \\ \boldsymbol{Y}_{0|i} \end{pmatrix} \end{aligned}$$

However, just as for the open-loop case, the effect of these different initial states dies out when the Kalman filters converge, i.e. when $i \to \infty$ [15, 18]. Based on this observation, we present a very simple but slightly biased algorithm in Figure 3.

This second algorithm is almost exactly the same as the algorithm in [15, 18]. The only differences are the following:

closed-loop algorithm 2:

1. Construct the matrices $M_{i|2i-1} = U_{i|2i-1} + H_i^c Y_{i|2i-1}$ and $M_{i+1|2i-1} = U_{i+1|2i-1} + H_{i-1}^c Y_{i+1|2i-1}$ where H_i^c and H_{i-1}^c contain the given Markov parameters of the controller.
2. Calculate the oblique projections:
$$\mathcal{O}_i = Y_{i|2i-1} /_{M_{i|2i-1}} \begin{pmatrix} \boldsymbol{U}_{0|i-1} \\ \mathbf{Y}_{0|i-1} \end{pmatrix} \quad , \quad \mathcal{O}_{i+1} = Y_{i+1|2i-1} /_{M_{i+1|2i-1}} \begin{pmatrix} \boldsymbol{U}_{0|i} \\ \mathbf{Y}_{0|i} \end{pmatrix}$$
3. Calculate the **SVD** of the oblique projection $\mathcal{O}_i = USV^T$.
4. Determine the order by inspecting the singular values in S and partition the **SVD** accordingly to obtain U_1, U_2 and S_1. Determine $\mathcal{G} = U_1 S_1^{1/2}$.
5. Determine the states:
$$\widetilde{X}_i = \mathcal{G}^\dagger \mathcal{O}_i \quad , \quad \widetilde{X}_{i+1} = (\underline{\mathcal{G}})^\dagger \mathcal{O}_{i+1} \ .$$
6. Proceed as in algorithm 1 (Figure 2, step 7) with $\widehat{X}_i$ and $\widehat{X}_{i+1}$ replaced by $\widetilde{X}_i$ respectively $\widetilde{X}_{i+1}$.

Figure 3: A schematic overview of the second closed-loop subspace identification algorithm. The bias goes to 0 as $i \to \infty$.

- In the oblique projections, use $M_{i|2i-1}$ and $M_{i+1|2i-1}$ instead of $U_{i|2i-1}$ and $U_{i+1|2i-1}$ in the open-loop case.
- The least squares solution does not determine A, B, C, D directly as in the open-loop case, but it determines the matrices in equation (49) instead. The closed-loop algorithm would even be more elegant when we could directly determine the matrices A, B, C, D from the states, but due to orthogonality problems (see the discussion under (46)-(48)) this is unfortunately not possible.

Apart from these two differences, the open and closed-loop algorithms are exactly the same. Unfortunately, these algorithms only compute asymptotically unbiased results when $i \to \infty$, which, in practical applications, is often not a very realistic assumption.

4.3 Algorithm 3: unbiased, using Γ_i

The two previous algorithms were based on the states of the system. It is however also possible to work with the shift-invariant structure of the extended observability matrix Γ_i [2, 20]. However, since we can only determine $T_i\Gamma_i$ from Theorem 2, we need to determine T_i before we can apply the shift-invariant property of Γ_i.

This is achieved by first determining $\mathcal{K}$ from (51) (see Appendix C). From $\mathcal{K}$ and the given Markov parameters of the controller, we then find T_i as:

$$T_i = I_{li} - \mathcal{K}H_i^c \ . \tag{58}$$

This implies that, with as before $\mathcal{G} = U_1 S_1^{1/2}$, the matrix Γ_i can be found as:

$$\Gamma_i = (I_{li} - \mathcal{K}H_i^c)^{-1}\mathcal{G} \ . \tag{59}$$

Once Γ_i is computed, the matrices A and C can be computed by any classical method making use of the shift-invariant structure [18] of Γ_i: least squares [2], total least squares [3], stable [1] or optimally [12]. From any of these, we find the matrices A and C.
The matrices B and D can then be found in many different ways. We only indicate one which is based on the projection algorithm of [2, 18]. From (33) we find:

$$U_2^T \mathcal{Z}_i = \underbrace{U_2^T (I_{li} - \mathcal{K}H_i^c)}_{\text{known}} .H_i^d . \underbrace{M_{i|2i+1}}_{\text{known}} \ ,$$

which is a linear equation in B and D.

Since $\mathcal{G} = T_i \Gamma_i$ is also highly structured, one could wonder if one couldn't directly determine A by shifting $\mathcal{G}$ up and down one block row. As is shown in Appendix D, this is not possible, since:

$$\underline{\mathcal{G}}^\dagger \overline{\mathcal{G}} = A - BD_cT_1C - \underline{\mathcal{G}}^\dagger T_{i-1} H_{i-1}^d \Gamma_{i-1}^c B_c T_1 C \ . \tag{60}$$

This is exactly the same as the result in (55). It thus seems that, without the explicit determination of $\mathcal{K}$, one always ends up with the expression (55) or (60).

4.4 Implementation

The implementation of the algorithms can be easily obtained with the RQ decomposition of [15, 18, 20]:

$$\frac{1}{\sqrt{j}} \begin{pmatrix} U_{0|2i-1} \\ Y_{0|2i-1} \end{pmatrix} = RQ^T$$

Once the R factor of this decomposition is computed, all data matrices (for instance $M_{i|2i-1}$) can be expressed as linear combinations of Q^T. We refer to [18, pp. 164] for expressions of the orthogonal and oblique projections in terms of the RQ decomposition. Observe that this "preprocessing" step represents a formidable data reduction and compression since the matrix Q is never explicitly needed.

4.5 Alternate controller configuration

One could wonder if the previous algorithms still hold if the controller is positioned in the feedforward loop as in Figure 4 (one degree of freedom controller) instead of the feedback loop as in Figure 1 (two degree of freedom controller). It can easily be proven that property (63) still holds for the matrix $N_{0|2i-1}$ as defined in (15). This is the key property in proving Theorems 1 and 2. Since both theorems are still valid, the three algorithms in the previous subsection will work just as well for the configuration in Figure 1 as for the configuration in Figure 4.

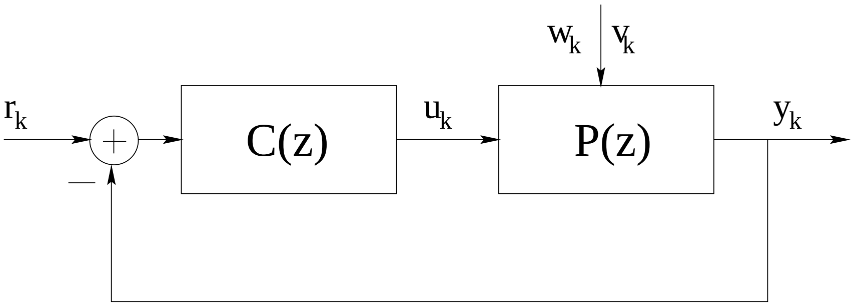

Figure 4: Alternate feedback setup. Compared to Figure 1, the controller has been moved to the feedforward path. However, Theorem 1 and 2 remain valid. As a consequence Algorithms 1 through 3 of this paper can still be applied for this controller configuration.

5 Example

In this section we illustrate the different algorithms discussed in this paper with a numerical simulation example.

The example is partially borrowed from [6] and is also used in [19]. The plant corresponds to a discrete time model of a laboratory plant setup of two circular plates rotated by an electrical servo motor with flexible shafts. In forward innovation state space form [18], the model becomes:

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + Ke_k \\ y_k &= Cx_k + Du_k + e_k \end{aligned}$$

with:

$$A = \begin{pmatrix} 4.40 & 1 & 0 & 0 & 0 \\ -8.09 & 0 & 1 & 0 & 0 \\ 7.83 & 0 & 0 & 1 & 0 \\ -4.00 & 0 & 0 & 0 & 1 \\ 0.86 & 0 & 0 & 0 & 0 \end{pmatrix}, B = \begin{pmatrix} 0.00098 \\ 0.01299 \\ 0.01859 \\ 0.0033 \\ -0.00002 \end{pmatrix}, C^T = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, K = \begin{pmatrix} 2.3 \\ -6.64 \\ 7.515 \\ -4.0146 \\ 0.86336 \end{pmatrix},$$

$D = 0$ and e_k a Gaussian white noise sequence with $\mathbf{E}[e_k e_k^T] = 1/9$. Note that the plant has one integrator and therefore is marginally stable. The configuration of model and controller is the one depicted in Figure 1. The controller has a state space description as in (5)-(6) with:

$$A_c = \begin{pmatrix} 2.65 & -3.11 & 1.75 & -0.39 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad , \quad B_c = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad , \quad C_c^T = \begin{pmatrix} -0.4135 \\ 0.8629 \\ -0.7625 \\ 0.2521 \end{pmatrix}$$

and $D_c = 0.61$. The reference signal r_k is a Gaussian white noise sequence with variance 1. We take the number of data points $j = 1200$ and the number of block rows $i = 10$. We generated 100 data sets, each time with the same reference input r_k but with a different noise sequence e_k. On each of these data sets, the following six algorithms were applied:

1. Algorithm 1 of this paper. Data given is u_k, y_k and the first 10 impulse responses of the controller.

2. Algorithm 2 of this paper. Data given is u_k, y_k and the first 10 impulse responses of the controller.

3. Algorithm 3 of this paper. Data given is u_k, y_k and the first 10 impulse responses of the controller.

4. The algorithm of [19]. The open-loop identification part is performed by the robust algorithm `subid` described in [18] (also with $i = 10$). We used balanced truncation as the model reduction technique in the second step. Data given is u_k,y_k and r_k.

5. The algorithm of [11] with the maximal order equal to 10 and the regression parameters $n_a = n_b = 11$. Data given is u_k and y_k.

6. The robust algorithm as described in [18], where we ignored the fact that the plant is working in closed-loop. Data given is u_k and y_k.

For each of the identified models, we computed the transfer function, the poles which are an indication of the goodness of fit of A and the eigenvalues of:

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix}, \tag{61}$$

which are an indication of the quality of fit of the the quadruple A, B, C, D.
The results are shown in Figures 5 through 7. From these figures we conclude the following:

1. Figure 5 shows the average (full line) and real (dashed line) transfer function for the six algorithms. Algorithm 1 clearly delivers the most accurate results. There is only a small bias in the first resonance peak and at high frequencies. Algorithm 3 and the algorithm of Verhaegen [19] deliver reasonably accurate results. Algorithm 2 has, as predicted in the theoretical derivation, a clear bias because i is finite. The algorithm of Ljung [11] is also biased, due to the finite n_a and n_b (which corresponds to a finite i). The last plot finally clearly indicates how an algorithm that works perfectly well in open-loop [18], completely breaks down in closed-loop.

   Finally note that all six algorithms deliver an accurate result around the cross over frequency of the system: 0.3 rad/sec.

2. Figure 6 shows for each of the algorithms the 100 estimates of the poles. The crosses indicate the true pole locations. Algorithm 1 and 3 compute accurate, unbiased results. Algorithm 2 and Ljung's algorithm compute results with a small variance, but with a slight bias. Even though Verhaegen's algorithm was shown to be unbiased in Figure 5, it is clear from Figure 6 that the variance is extremely high. This is because first a ninth order system is estimated, the order of which is then reduced. This complex procedure clearly introduces quite some extra variance. The poles calculated from the open-loop algorithm are completely wrong at low frequencies.

3. Finally, Figure 7 shows the estimated eigenvalues of (61). The same conclusions as for Figure 6 hold, except that now also algorithm 3 has a larger variance.

From these three figures, we can conclude that Algorithm 1 is, for this example, superior to the other algorithms. It computes accurate results with a low variance on transfer function, poles and eigenvalues of (61).

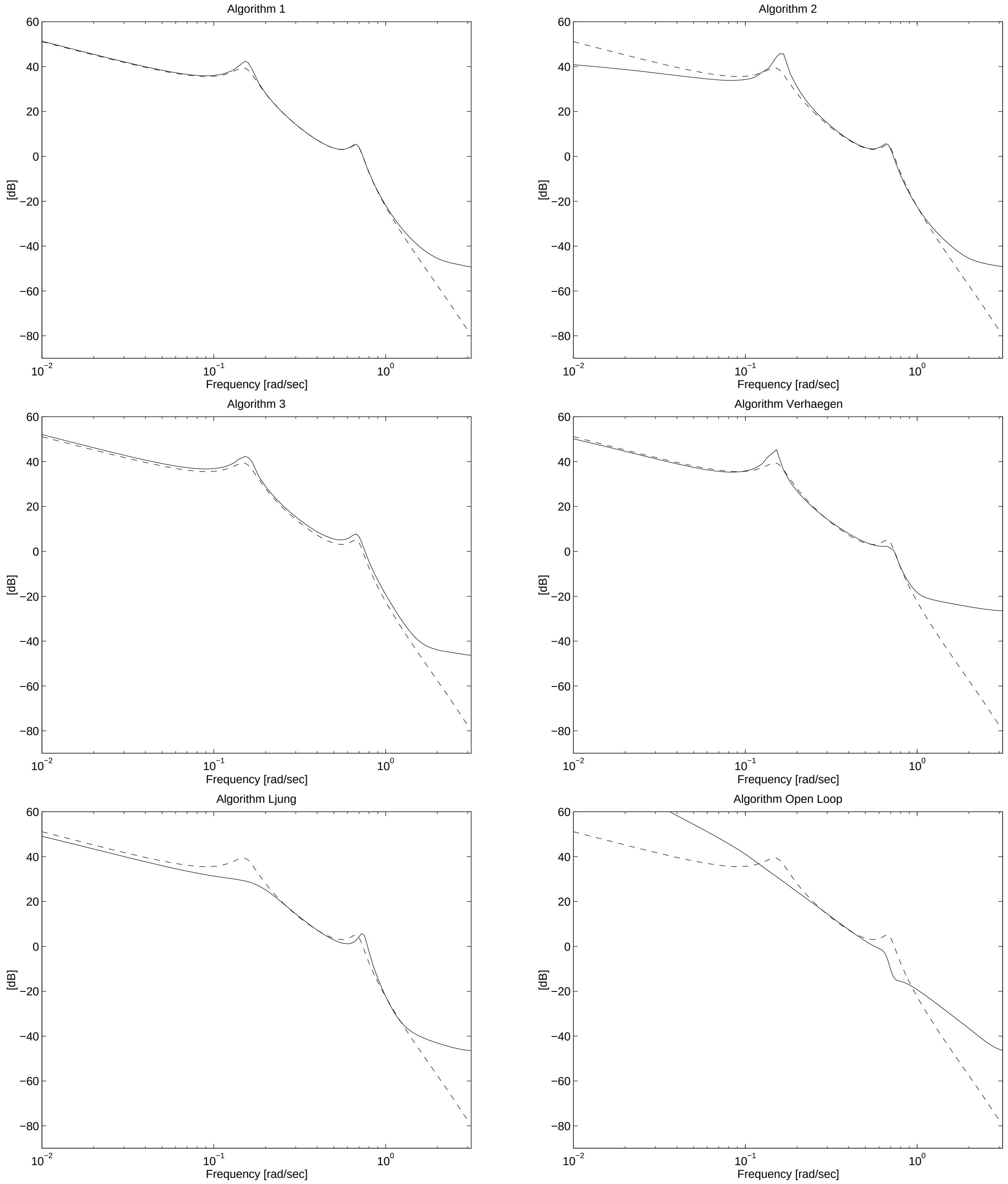

Figure 5: True transfer function (dashed line) and average transfer function over 100 experiments (full line). Algorithm 1,3 and the algorithm of Verhaegen deliver accurate results. Algorithm 2, Ljung's algorithm and the open-loop algorithm compute biased results.

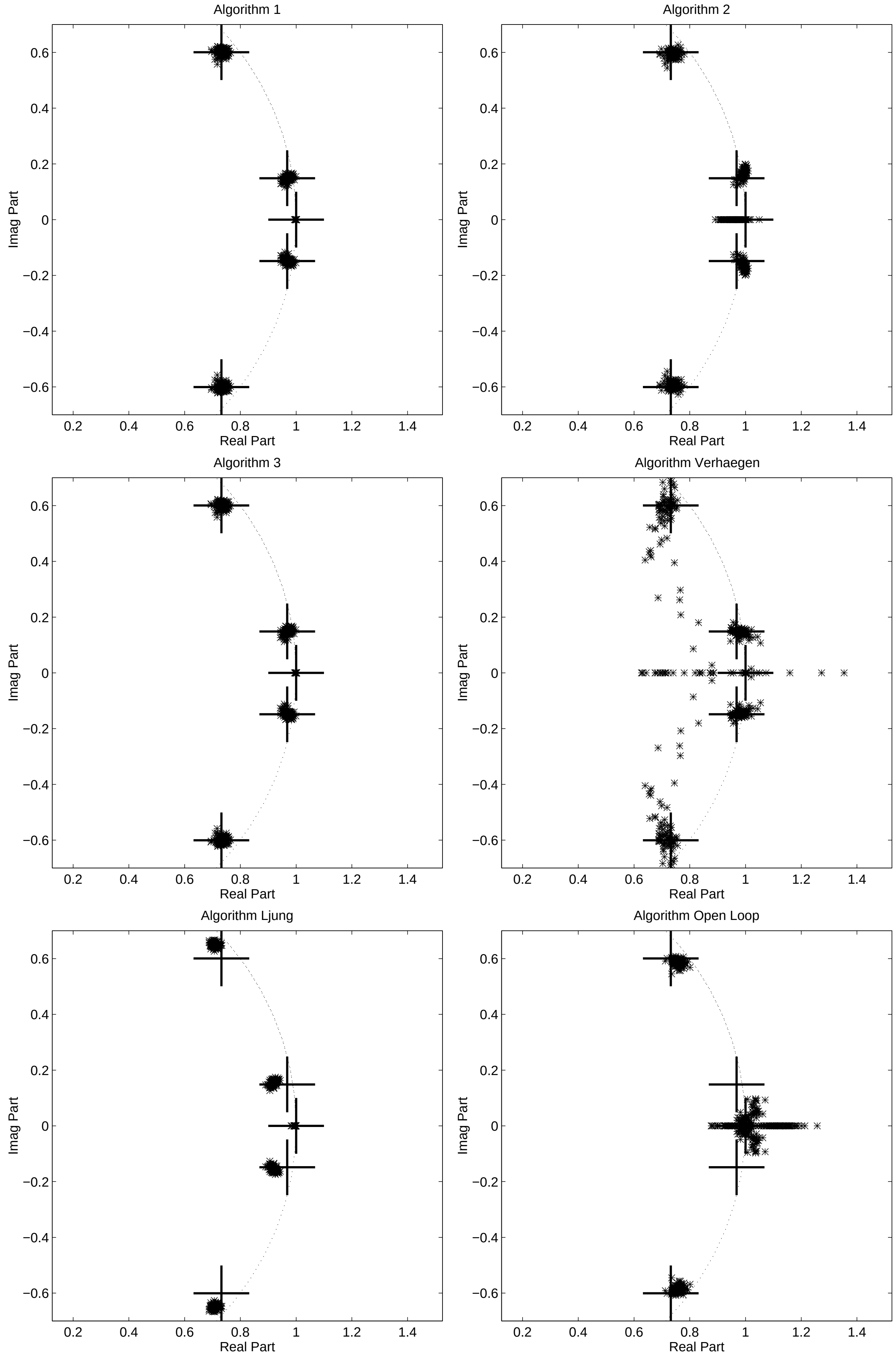

Figure 6: Pole estimates over 100 experiments. The crosses indicate the true pole locations. Algorithm 1 and 3 compute accurate, unbiased results. Algorithm 2 and Ljung's algorithm have a slight systematic error. The poles of Verhaegen's algorithm have a large variance, while the open-loop algorithm computes totally wrong poles at low frequencies.

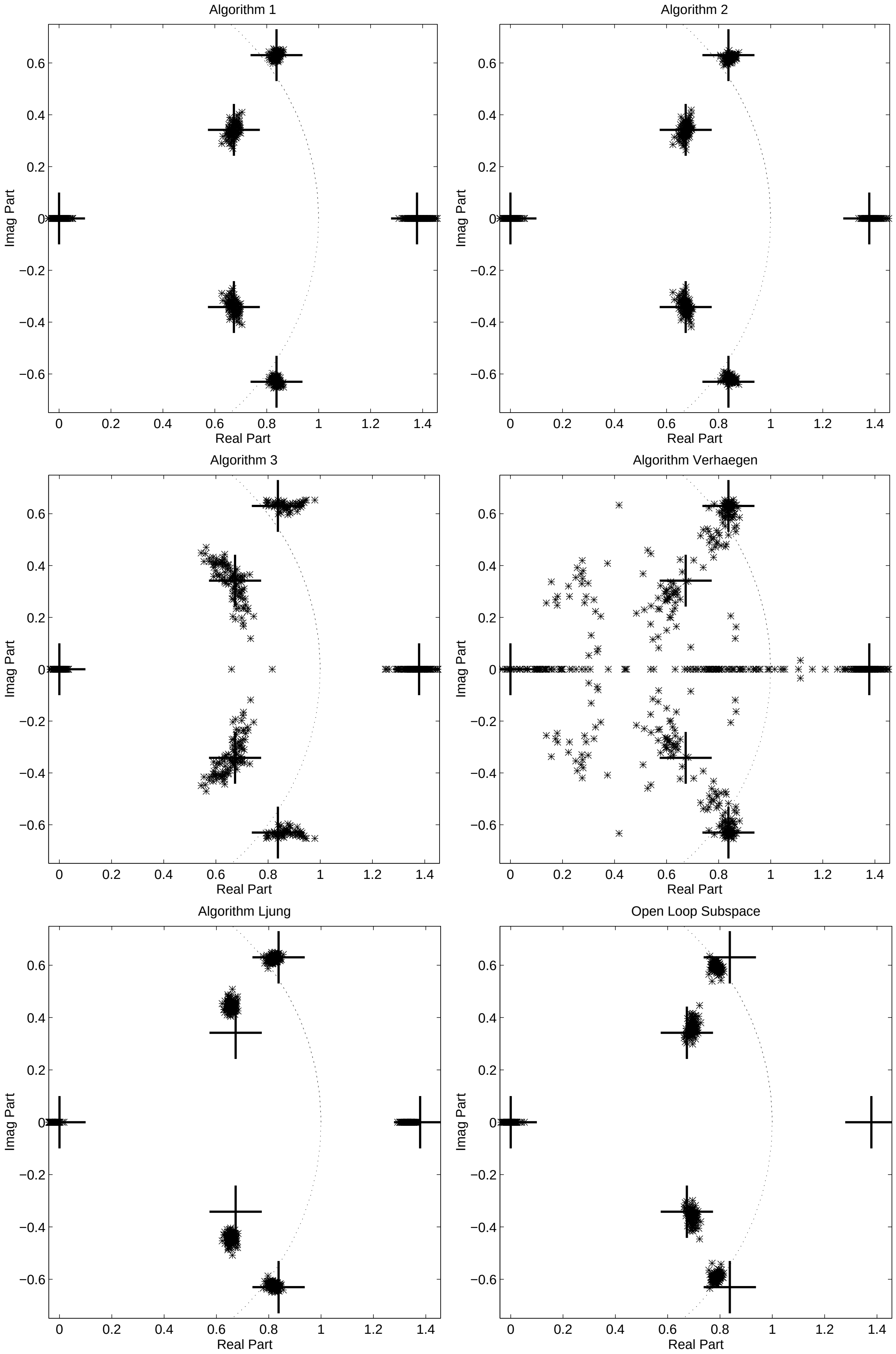

Figure 7: eigenvalues of (61). The same conclusions of Figure 6 hold except that now also algorithm 3 computes higher variance results.

Finally, looking at the computational load, we found that Algorithms 1 through 3 and the open-loop identification algorithm have a similar computational load. The algorithm of Verhaegen has a load of a factor 2.5 higher. This is of course due to the fact that the input and output are jointly identified. Finally, the algorithm of Ljung has a complexity of a factor 10 higher compared to algorithms 1 through 3. This could also be due to the non-optimal implementation (we just copied the Matlab description in [11]). The loads were computed from the number of floating point operations indicated in Matlab.

6 Conclusions

In this paper we have derived a general framework for subspace identification of systems working in closed-loop. Two new theorems that allow for the reconstruction of the states and for the recovery of system related matrices from input-output data have been derived. These theorems are very similar to the classical open-loop subspace theorems, to which they are reduced when there is no controller active in the loop. From these theorems, three subspace identification algorithms have been derived and asymptotically analyzed. An example was used to illustrate the power and properties of the closed-loop subspace identification algorithms.

The closed-loop subspace identification algorithms presented in this paper can also be presented in the unifying framework of [16, 18]. In this way, **MOESP** [20] and **CVA** [8] can be made consistent for closed-loop measurements. Furthermore we can also interpret the results in the framework of the frequency weighted model reduction [17, 18], which leads to nice interpretations when identifying lower order models. Treating these properties in this paper would lead us too far, so they will be the topic of a follow-up paper.

References

[1] Chui N.L.C., Maciejowski J.M. *Realization of Stable Models with Subspace Methods.* Proc. of the World Congress of the International Federation of Automatic Control, IFAC, Vol. **I**, pp. 327-332, San Francisco, USA, 1996.

[2] De Moor B., Vandewalle J. *A geometrical strategy for the identification of state space models of linear multivariable systems with singular value decomposition.* Proc. of the 3rd International Symposium on Applications of Multivariable System Techniques, April 13-15, Plymouth, UK, pp. 59-69, 1987.

[3] De Moor B. *Mathematical concepts and techniques for modeling of static and dynamic systems.* PhD thesis, Department of Electrical Engineering, Katholieke Universiteit Leuven, Belgium, 1988.

[4] De Moor B., Van Overschee P. *Numerical Algorithms for Subspace State Space System Identification.* Trends in Control. A European Perspective. Ed. A. Isidori, European Control Conference, Italy, pp. 385-422, 1995.

[5] Gustavsson I., Ljung L., Söderström T. *Identification of Processes in Closed-Loop - Identifiability and Accuracy Aspects.* Automatica, Vol. **13**, pp. 59-75, 1977.

[6] Hakvoort R. *Approximate Identification in the controller design problem.* Master Thesis Delft University of Technology, The Netherlands, Measurement and Control Theory Section, Mech, Eng., **A-538**, 1990.

[7] Kailath T. *Linear Systems.* Prentice Hall, Englewood Cliffs, New Jersey, 1980.

[8] Larimore W.E. *Canonical variate analysis in identification, filtering and adaptive control.* Proc. 29th Conference on Decision and Control, Hawai, USA, pp. 596-604, 1990.

[9] Larimore W.E. *Optimal Order Selection and Efficiency of Canonical Variate Analysis System Identification.* Proc. of the World Congress of the International Federation of Automatic Control, IFAC, Vol. **I**, pp. 151-156, San Francisco, USA, 1996.

[10] Ljung L. *System identification - Theory for the User.* Prentice Hall, Englewood Cliffs, NJ, 1987.

[11] Ljung L., McKelvey T. *Subspace Identification from Closed Loop Data.* Signal Processing, Special Issue on Subspace Methods, Part II: System Identification, Vol. **52**, No. 2, pp. 209-216, 1996. (See also Internal Report LiTH-ISY-R-1752, Department of electrical engineering, Linköping University, Sweden, 1995 for the Matlab code).

[12] Ottersten B., Viberg M., *A Subspace Based Instrumental Variable Method for State Space System Identification.* Proc. of SYSID '94, Vol. **2**, 4-6 July, Copenhagen, Denmark, pp.139-144, 1994.

[13] Söderström T., Stoica P. *System Identification.* Prentice Hall International Series in Systems and Control Engineering, Prentice Hall, New York, 1989.

[14] Van Overschee P., De Moor B. *Subspace algorithms for the stochastic identification problem.* Automatica, Vol. **29**, no. 3, pp. 649-660, 1993.

[15] Van Overschee P., De Moor B. *N4SID: Subspace Algorithms for the Identification of Combined Deterministic-Stochastic Systems.* Automatica, Special Issue on Statistical Signal Processing and Control, Vol. **30**, no. 1, pp. 75-93, 1994.

[16] Van Overschee P., De Moor B. *A Unifying Theorem for Three Subspace System identification Algorithms.* Automatica, Vol. **31**, no. 12, pp. 1853-1864, 1995.

[17] Van Overschee P., De Moor B. *Choice of State Space Basis in Combined Deterministic-Stochastic Subspace Identification.* Automatica, Vol. **31**, no. 12, pp. 1877-1883, 1995.

[18] Van Overschee P., De Moor B. *Subspace Identification for Linear Systems. Theory, Implementation, Applications.* Kluwer Academic Press, Dordrecht, 1996.

[19] Verhaegen M. *Application of a Subspace Model Identification Technique to Identify LTI Systems Operating in Closed-Loop.* Automatica, Vol. **29**, no. 4, pp. 1027-1040, 1993.

[20] Verhaegen M. *Identification of the deterministic part of MIMO state space models given in innovations form from input-output data*, Automatica, Special Issue on Statistical Signal Processing and Control, Vol **30**, no. 1, pp. 61-74, 1994.

[21] Viberg M. *Subspace Methods in System Identification.* Proc. of SYSID '94, Vol. **1**, 4-6 July, Copenhagen, Denmark, pp. 1-12, 1994.

A Proof of the closed-loop Theorem

The proof of (32) is trivial, since (using (16) and (17)):

$$\begin{pmatrix} N_{0|2i-1} \\ Y_{0|i-1} \end{pmatrix} = \begin{pmatrix} N_{0|i-1} \\ N_{i|2i-1} \\ Y_{0|i-1} \end{pmatrix} = \begin{pmatrix} I_{mi} & 0 & H_i^c \\ 0 & I_{mi} & \Gamma_i^c \Delta_i^c \\ 0 & 0 & I_{li} \end{pmatrix} \begin{pmatrix} U_{0|i-1} \\ M_{i|2i-1} \\ Y_{0|i-1} \end{pmatrix} . \tag{62}$$

The left factor in (62) is square upper triangular and has only ones on the diagonal. It is thus of full rank which implies that:

$$\text{row space} \begin{pmatrix} N_{0|2i-1} \\ Y_{0|i-1} \end{pmatrix} = \text{row space} \begin{pmatrix} U_{0|i-1} \\ M_{i|2i-1} \\ Y_{0|i-1} \end{pmatrix} ,$$

which proves (32).

□

Before we start the bulk of the proof of (33), we first prove the following crucial properties:

$$\begin{aligned} \mathbf{E_j}\left[(\Gamma_{2i} X_0^d + Y_{0|2i-1}^s) N_{0|2i-1}^T\right] &= \Gamma_{2i} S_{xn} , & (63) \\ \mathbf{E_j}[Y_{0|2i-1}^s (X_0^d)^T] &= \Gamma_{2i} \Sigma_{sd} . & (64) \end{aligned}$$

The input-output equation corresponding to $Y_{0|2i-1}^s$ is (the stochastic part of (1)-(2)):

$$Y_{0|2i-1}^s = \Gamma_{2i} X_0^s + \underbrace{\begin{pmatrix} 0 & 0 & \dots & 0 \\ C & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ CA^{i-2} & CA^{i-3} & \dots & 0 \end{pmatrix}}_{=H_{2i}^w} W_{0|2i-1} + V_{0|2i-1}$$

So (63) can be rewritten as:

$$\mathbf{E_j}\left[(\Gamma_{2i} X_0 + H_{2i}^w W_{0|2i-1} + V_{0|2i-1}) N_{0|2i-1}^T\right] . \tag{65}$$

It is also easy to show (from (15)) that $N_{0|2i-1} = R_{0|2i-1} - \Gamma_{2i}^c X_0^c$, which implies that:

$$\begin{aligned} \mathbf{E_j}\left[W_{0|2i-1} N_{0|2i-1}^T\right] &= \mathbf{E_j}\left[W_{0|2i-1} R_{0|2i-1}^T + W_{0|2i-1} (X_0^c)^T (\Gamma_{2i}^c)^T\right] \\ &= \underbrace{\mathbf{E_j}[W_{0|2i-1} R_{0|2i-1}^T]}_{=0} + \underbrace{\mathbf{E_j}[W_{0|2i-1} (X_0^c)^T]}_{=0} (\Gamma_{2i}^c)^T \\ &= 0 , & (66) \end{aligned}$$

where the first term is zero due to assumption 1 of Theorem 1 and the second term due to w_k being white noise. A property similar to (66) holds for $V_{0|2i-1}$. Combining (65) and (66) and the definition of S_{xn} in (23) proves (63). The proof of (64) is very similar.

oxo

□

The proof of (33) is now very similar to the proof in [15, 18]. We thus skip the more trivial steps, and refer the reader to [15, 18] for more details. We first rewrite $Y_{i|2i-1}$:

$$\begin{aligned}
Y_{i|2i-1} &= \Gamma_i X_i^d + H_i^d U_{i|2i-1} + Y_{i|2i-1}^s \\
&= \Gamma_i \left[A^i X_0^d + \Delta_i^d U_{0|i-1}\right] + H_i^d \left[M_{i|2i-1} - H_i^c Y_{i|2i-1}\right] + Y_{i|2i-1}^s \\
\underbrace{(I_{li} + H_i^d H_i^c)}_{=T_i^{-1}} Y_{i|2i-1} &= \Gamma_i A^i X_0^d + Y_{i|2i-1}^s + \Gamma_i \Delta_i^d U_{0|i-1} + H_i^d M_{i|2i-1} \\
Y_{i|2i-1} &= T_i \left[\underbrace{\Gamma_i A^i X_0^d + Y_{i|2i-1}^s}_{\text{Term 1}} + \underbrace{\Gamma_i \Delta_i^d U_{0|i-1} + H_i^d M_{i|2i-1}}_{\text{Term 2}}\right] . \qquad (67)
\end{aligned}$$

When we now project $Y_{i|2i-1}$ as in (32) we see that “Term 2” of (67) does not change since “Term 2” lies in the row space we are projecting on. We thus only need to compute the projection of “Term 1”. Hereto we define as in [15, 18]:

$$\begin{aligned}
\mathcal{A} &= \mathbf{E_j}\left[(\Gamma_i A^i X_0^d + Y_{i|2i-1}^s)\left(\begin{array}{c|c} N_{0|2i-1}^T & Y_{0|i-1}^T \end{array}\right)\right] = \left(\begin{array}{c|c} \mathcal{A}_1 & \mathcal{A}_2 \end{array}\right) \\
\mathcal{B} &= \mathbf{E_j}\left[\left(\frac{N_{0|2i-1}}{Y_{0|i-1}}\right)\left(\begin{array}{c|c} N_{0|2i-1}^T & Y_{0|i-1}^T \end{array}\right)\right] = \left(\begin{array}{c|c} \mathcal{B}_{11} & \mathcal{B}_{21}^T \\ \hline \mathcal{B}_{21} & \mathcal{B}_{22} \end{array}\right)
\end{aligned}$$

With a derivation similar to (67) we find that

$$Y_{0|i-1} = T_i \left[\Gamma_i X_0^d + H_i^d N_{0|i-1} + Y_{0|i-1}^s\right] ,$$

which leads to:

$$\begin{aligned}
\mathcal{A}_1 &= \Gamma_i A^i S_{xn} \\
\mathcal{A}_2 &= \left[\Gamma_i A^i (\Sigma_{dd} + \Sigma_{sd} + \Sigma_{sd}^T)\Gamma_i^T + \Gamma_i A^i S_{xn} + \Gamma_i \Delta_i^G\right] T_i^T \\
\mathcal{B}_{11} &= R_{nn} \\
\mathcal{B}_{21} &= T_i \left[\Gamma_i A^i S_{xn} + H_i^d \begin{pmatrix} I & 0 \end{pmatrix} R_{nn}\right] \\
\mathcal{B}_{22} &= T_i \Big[\Gamma_i (\Sigma_{dd} + \Sigma_{sd} + \Sigma_{sd}^T)\Gamma_i^T + L_i \\
&\quad + \Gamma_i S_{xn} \begin{pmatrix} I \\ 0 \end{pmatrix} (H_i^d)^T + H_i^d \begin{pmatrix} I & 0 \end{pmatrix} S_{xn}^T \Gamma_i^T + H_i^d \begin{pmatrix} I & 0 \end{pmatrix} R_{nn} \begin{pmatrix} I \\ 0 \end{pmatrix} (H_i^d)^T \Big] T_i^T
\end{aligned}$$

Using the matrix inversion lemma [7] and proceeding as in [15, 18], and with Ω_i defined as in (27) and P_0 and $\widehat{X}_0$ as in (35)-(36) we finally find that:

$$\begin{aligned}
\mathcal{Z}_i &= T_i \mathcal{A} \mathcal{B}^{-1} \begin{pmatrix} N_{0|2i-1} \\ Y_{0|i-1} \end{pmatrix} + T_i \Gamma_i \Delta_i^d U_{0|i-1} + T_i H_i^d M_{i|2i-1} \\
&= T_i \Gamma_i \left[(A^i - \Omega_i \Gamma_i) \underbrace{S_{xn} R_{nn}^{-1} N_{0|2i-1}}_{=\widehat{X}_0} + \Omega_i \underbrace{(T_i^{-1} Y_{0|i-1} - H_i^d N_{0|i-1})}_{=\Gamma_i X_0^d + Y_i^s = Y_{0|i-1} - H_i^d U_{0|i-1}} + \Delta_i^d U_{0|i-1}\right] + T_i H_i^d M_{i|2i-1} \\
&= T_i \Gamma_i \left[(A^i - \Omega_i \Gamma_i)\widehat{X}_0 + (\Delta_i^d - \Omega_i H_i^d) U_{0|i-1} + \Omega_i Y_{0|i-1}\right] + T_i H_i^d M_{i|2i-1} ,
\end{aligned}$$

which proves (33).

□

B Proof of the orthogonal residuals

We first show that the row-space of $E_{i|i}$ is orthogonal to the row spaces of the states $\widehat{X}_i$ and of the matrix $M_{i|i}$. From the first block row of (33) we find that:

$$Y_{i|i} = T_1 C\widehat{X}_i + T_1 DM_{i|i} + T_1\rho \ . \tag{68}$$

where $T_1\rho$ is the residual of projecting $Y_{i|2i-1}$ onto the row space of $N_{0|2i-1}$ and $Y_{0|i-1}$. This implies that (since T_1 is of full rank) the row space of ρ is orthogonal to the row spaces of $\widehat{X}_i$ and $M_{i|i}$. Indeed, $\widehat{X}_i$ and $M_{i|i}$ can be written as a linear combination of $N_{0|2i-1}$ and $Y_{0|i-1}$ (see (17) and the fact that $\widehat{X}_i$ is a linear combination of $\widehat{X}_0$, $U_{0|i-1}$ and $Y_{0|i-1}$).
Now we prove that the residual ρ equals $E_{i|i}$. From (68) we find that:

$$\begin{aligned} \rho &= T_1^{-1}Y_{i|i} - C\widehat{X}_i - DM_{i|i} \\ &= (I_l + DD_c)Y_{i|i} - C\widehat{X}_i - D(U_{i|i} + D_cY_{i|i}) \\ &= Y_{i|i} - C\widehat{X}_i - DU_{i|i} \\ &= E_{i|i} \ , \end{aligned}$$

where the last step follows from (48).
Finally we show that (49) holds true. From (47) and the fact that $U_{i|i} = M_{i|i} - D_cY_{i|i}$ we find that:

$$\begin{aligned} Y_{i|i} &= C\widehat{X}_i + D(M_{i|i} - D_cY_{i|i}) + E_{i|i} \ , \\ \underbrace{(I_l + DD_c)}_{=T_1^{-1}} Y_{i|i} &= C\widehat{X}_i + DM_{i|i} + E_{i|i} \ , \\ Y_{i|i} &= T_1C\widehat{X}_i + T_1DM_{i|i} + T_1E_{i|i} \ . \end{aligned} \tag{69}$$

The last equation is the bottom equation of (49). From (46) and (69) we then find:

$$\begin{aligned} \widehat{X}_{i+1} &= A\widehat{X}_i + B(M_{i|i} - D_cY_{i|i}) + K_iE_{i|i} \\ &= A\widehat{X}_i + BM_{i|i} - BD_c(T_1C\widehat{X}_i + T_1DM_{i|i} + T_1E_{i|i}) + K_iE_{i|i} \\ &= (A - BD_cT_1C)\widehat{X}_i + B(I_m - D_cT_1D)M_{i|i} + (K_i - BD_cT_1)E_{i|i} \ , \end{aligned}$$

which is the top equation of (49).

□

C Solution of (51)

Define:

$$\begin{aligned} U_2^T &= \begin{pmatrix} \mathcal{U}_0 & \mathcal{U}_1 & \ldots & \mathcal{U}_{i-1} \end{pmatrix} \ , \\ M_{i|2i-1} &= \begin{pmatrix} \mathcal{M}_0 \\ \mathcal{M}_1 \\ \ldots \\ \mathcal{M}_{i-1} \end{pmatrix} \ , \end{aligned}$$

where $\mathcal{U}_k \in \mathbb{R}^{(li-n)\times l}$ and $\mathcal{M}_k \in \mathbb{R}^{m\times j}$. We can rewrite (51) as:

$$\begin{aligned} U_2^T \mathcal{Z}_i &= \begin{pmatrix} \mathcal{U}_0 & \dots & \mathcal{U}_{i-1} \end{pmatrix} \begin{pmatrix} \mathcal{K}_0 \\ \dots \\ \mathcal{K}_{i-1} \end{pmatrix} \mathcal{M}_0 + \begin{pmatrix} \mathcal{U}_1 & \dots & \mathcal{U}_{i-1} \end{pmatrix} \begin{pmatrix} \mathcal{K}_0 \\ \dots \\ \mathcal{K}_{i-2} \end{pmatrix} \mathcal{M}_1 \\ & \quad + \dots + \mathcal{U}_{i-1}\mathcal{K}_0\mathcal{M}_{i-1} \end{aligned}$$

Using the property that $\mathrm{vec}(AXB) = (B^T \otimes A)\mathrm{vec}X$, where $\mathrm{vec}A$ denotes the vector operation and $\otimes$ the Kronecker product, we find:

$$\begin{aligned} \mathrm{vec}(U_2^T \mathcal{Z}_i) &= \left[\mathcal{M}_0^T \otimes \begin{pmatrix} \mathcal{U}_0 & \dots & \mathcal{U}_{i-1} \end{pmatrix}\right] \mathrm{vec} \begin{pmatrix} \mathcal{K}_0 \\ \dots \\ \mathcal{K}_{i-1} \end{pmatrix} \\ & \quad + \left[\mathcal{M}_1^T \otimes \begin{pmatrix} \mathcal{U}_1 & \dots & \mathcal{U}_{i-1} \end{pmatrix}\right] \mathrm{vec} \begin{pmatrix} \mathcal{K}_0 \\ \dots \\ \mathcal{K}_{i-2} \end{pmatrix} \\ & \quad + \dots + \left[\mathcal{M}_{i-1}^T \otimes \mathcal{U}_{i-1}\right] \mathrm{vec}\mathcal{K}_0 \ . \end{aligned} \tag{70}$$

This last equation is a linear equation in the unknowns $\mathcal{K}_k$. There are $(li-n)j$ equations in ilm unknowns. With the use of the RQ decomposition of Subsection 4.4, the number of equations can be brought back to $2i(li-n)(l+m)$.

D Additional Proofs

Proof of (55)-(56).

Starting from equation (49) and substituting $\widehat{X}_i$ and $\widehat{X}_{i+1}$ in that equation by (52) and (53) we find:

$$\begin{pmatrix} \underline{\mathcal{G}}^\dagger \mathcal{Z}_{i+1} \\ Y_{i|i} \end{pmatrix} = \begin{pmatrix} A - BD_cT_1C \\ T_1C \end{pmatrix} \mathcal{G}^\dagger \mathcal{Z}_i + \begin{pmatrix} \underline{\mathcal{G}}^\dagger T_{i-1} H_{i-1}^d \\ 0 \end{pmatrix} M_{i+1|2i-1} + \bullet M_{i|2i-1} \ , \tag{71}$$

where the $\bullet$ indicates a don't care. Substituting the bottom expression for $Y_{i|i}$ in (18) leads to:

$$M_{i+1|2i-1} = \overline{M}_{i|2i-1} - \Gamma_{i-1}^c B_c T_1 C \mathcal{G}^\dagger \mathcal{Z}_i + \bullet M_{i|2i-1} \ .$$

Substitution of this last equation in (71) finally leads to:

$$\begin{aligned} \begin{pmatrix} \underline{\mathcal{G}}^\dagger \mathcal{Z}_{i+1} \\ Y_{i|i} \end{pmatrix} &= \begin{pmatrix} A - BD_cT_1C \\ T_1C \end{pmatrix} \mathcal{G}^\dagger \mathcal{Z}_i - \begin{pmatrix} \underline{\mathcal{G}}^\dagger T_{i-1} H_{i-1}^d \\ 0 \end{pmatrix} \Gamma_{i-1}^c B_c T_1 C \mathcal{G}^\dagger \mathcal{Z}_i + \bullet M_{i|2i-1} \\ &= \begin{pmatrix} A - BD_cT_1C - \underline{\mathcal{G}}^\dagger T_{i-1} H_{i-1}^d \Gamma_{i-1}^c B_c T_1 C \\ T_1C \end{pmatrix} \mathcal{G}^\dagger \mathcal{Z}_i + \bullet M_{i|2i-1} \ , \end{aligned}$$

which proves (55)-(56).

Proof of (60).

Using the block matrix inversion lemma of [7], and the definitions of all matrices involved, we find:

$$
\begin{aligned}
\overline{\mathcal{G}} &= \overline{T_i}\Gamma_i \\
&= \overline{[(I_{li} + H_i^d H_i^c)^{-1}]}\Gamma_i \\
&= \overline{\left[\begin{pmatrix} T_1^{-1} & 0 \\ \Gamma_{i-1}BD_c + H_{i-1}^d\Gamma_{i-1}^c B_c & T_{i-1}^{-1} \end{pmatrix}^{-1}\right]}\Gamma_i \\
&= \overline{\begin{pmatrix} T_1 & 0 \\ -T_{i-1}\Gamma_{i-1}BD_cT_1 - T_{i-1}H_{i-1}^d\Gamma_{i-1}^c B_c T_1 & T_{i-1} \end{pmatrix}}\begin{pmatrix} C \\ \Gamma_{i-1}A \end{pmatrix} \\
&= T_{i-1}\Gamma_{i-1}A - T_{i-1}\Gamma_{i-1}BD_cT_1C - T_{i-1}H_{i-1}^d\Gamma_{i-1}^c B_c T_1 C \\
&= \underline{\mathcal{G}}(A - BD_cT_1C) - T_{i-1}H_{i-1}^d\Gamma_{i-1}^c B_c T_1 C \ ,
\end{aligned}
$$

which proves (60).

□